

Packages Cdn

All Play+ packages are distributed exclusively through your organization's private npm registry . This ensures you are working with secure, vetted, and version-controlled assets aligned with internal governance policies.

Public CDN Not Available

To maintain strict version control and security compliance, Play+ does not offer a public CDN . All assets—including component libraries, stylesheets, and icons—must be installed as npm packages from the configured private registry.

This guarantees your applications are built with enterprise-maintained, audited code , supporting traceability and repeatability across environments.

Using Play+ without a Framework

Play+ can be used to build static websites or prototypes without a JavaScript framework. This is especially useful for documentation sites, rapid prototyping, or UI testing.

The following guide outlines how to set up Play+ with plain HTML, CSS, and JavaScript .

Quickstart: Static HTML Setup

1. Create a Project Directory

Initialize a new directory and package.json :

2. Install the Core UI Package

Install the core Play+ UI package from your private registry. This includes precompiled CSS, JavaScript modules, and web components.

3. Create the HTML File

Create a basic index.html file in your project root. This file will link to the Play+ assets and contain your markup.

4. Link to Play+ Assets

Add `<link>` and `<script>` tags that point to the required assets from the `node_modules` folder.

Using Icons with Play+

Play+ integrates the Lucide icon set—a pixel-perfect, open-source icon system known for its consistency and clarity.

Icons are packaged as Web Components via the `@playplus/icons` package, which is automatically installed as a dependency of `@playplus/ui`.

You can use icons directly in your HTML with the `<pp-icon>` tag:

Browse the full list of available icons on the [Lucide icon directory](#).

Summary

- Play+ packages are available only via the internal npm registry.
- Public CDN is intentionally disabled to enforce enterprise security and consistency.
- Framework-less usage is supported via native Web Components and static asset linking.
- Lucide-based icons are fully integrated and require no additional installation.
- For plain HTML setups, remember to resolve `node_modules` assets using a copy or static server workflow.